OA 4202, Homework 1

Nedialko B. Dimitrov

1. In class, we ran BFS on the graph in Figure 1. When we had a choice of multiple "white" neighbors to push on to Q, in class, we always pushed the lowest numbered neighbor first. Re-do BFS on the same graph, but this time, always push the highest numbered neighbor first. Draw the resulting BFS search tree. Is the resulting search tree the same as the one we saw in class?

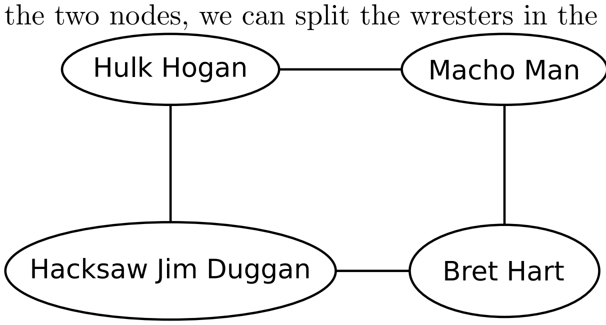
2. In class, we ran DFS on the graph in Figure 1. When we had a choice of multiple "white" neighbors to push on to Q, in class, we always pushed the lowest numbered neighbor first. Re-do DFS on the same graph, but this time, always push the highest numbered neighbor first. Draw the resulting DFS search tree. Is the resulting search tree the same as the one we saw in class?

3. What is the running time of BFS if the input graph is stored using a dense adjacency matrix?

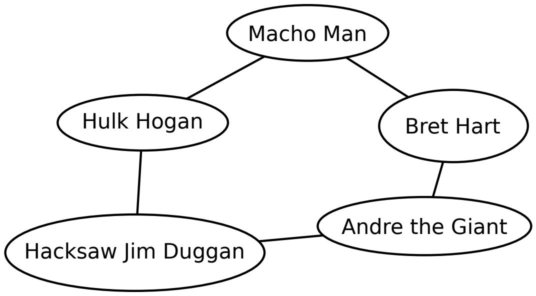
4. You are dropped in the middle of a maze full of hallways and intersections. Everything looks the same to you. Every intersection looks exactly like the previous, every hallway looks exactly like the next. Luckily, you know there is an exit and you happen to have an infinite supply of pennies. How do you get out?

5. Between any pair of professional wrestlers, there may or may not be a rivalry. For marketing purposes, it is helpful to divide professional wrestlers in two separate groups: "good guys" and "bad guys." Suppose we have n professional wrestlers and a list of m rivalries between pairs of wrestlers. Describe an algorithm that runs in $O(n+m)$ time and determines whether it is possible to split the professional wrestlers into "good guys" and "bad guys," so that each rivalry is between a good guy and a bad guy. If such a labeling exists, your algorithm should produce it.

   For example, if each wrestler is a node and a rivalry between two wrestlers is an edge between the two nodes, we can split the wresters in the following graph

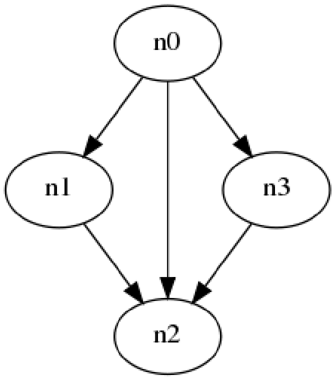
   

   by making Hulk Hogan and Bret Hart "good guys," and Macho Man and Hacksaw Jim Duggan "bad guys." On the other hand, in the following graph, it is impossible to do the splitting



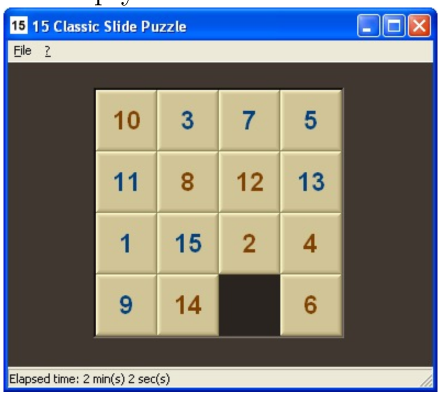
(adapted from CLRS)

6. Give an $O(n + m)$ time algorithm that takes as input a directed acyclic graph $G = (V, E)$ and two nodes s and t, and returns the number of paths from s to t in G. For example, this graph

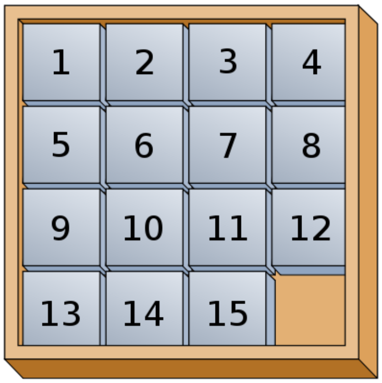
   has three paths from 0 to 2 ([0,2], [0,1,2], and [0,3,2]). (Hint: Check your algorithm by making sure it works on a few small examples.) (adapted from CLRS).

7. Recall the classic "15 puzzle" game. In this game, we are given a board with 15 pieces and one empty slot. The board looks something like this:

   We are only allowed to slide tiles into the empty slot, leaving a new empty slot where the tile used to be. For example, in the above picture, we can slide the "2" down into the empty slot, leaving a new empty slot on row 2, column 2. Similarly, we can slide the "6" left into the slot or the "14" right into the slot. Those three moves are the only moves allowed from that position of the board.

   Our goal is to get the board to look like this:



Suppose we are given an arbitrary board position and are asked to solve the puzzle. First, formulate a graph reachability problem that tells us whether the puzzle is solvable. Second, if the puzzle is solvable, how could we use an algorithm we learned in class to solve the puzzle with a minimum number of moves? (adapted from AMO)

8. Suppose you only have acces to BFS, and cannot change the way the algorithm works, only its input. You are given a graph with positive integer edge lengths. How can you find the shortest path between two nodes, s and t, using BFS.

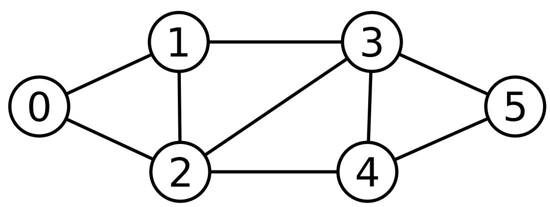


Figure 1: The example graph from class, on which we ran BFS and DFS.